On Metaheuristic "Failure Modes": A Case Study in Tabu Search for Job-Shop Scheduling *

Jean-Paul Watson*

*Department of Discrete Algorithms and Mathematics, Sandia National Laboratories
P.O. Box 5800, MS 1110, Albuquerque, NM 87185-1110 USA
{jwatson}@sandia.gov

1 Introduction

It is widely recognized that long-term memory mechanisms such as reintensification and path relinking can significantly improve the performance of "core" metaheuristics such as tabu search or iterated local search. Case studies abound, e.g., see [2]. These algorithms are commonly designed around a set or pool of elite, high-quality solutions from which intensification and diversification schemes are initiated. However, it is equally well known that the details of how the elite pool is managed has a major impact on performance [3]. Considering that the relationships between elite pool maintenance, long-term memory, and search space structure are poorly understood, the fact that the development of effective long-term memory and pool maintenance mechanisms is more art than science is not surprising. In this paper, we analyze the relationship between pool maintenance schemes, long-term memory mechanisms, and search space structure, with the goal of placing metaheuristic design on a more concrete foundation.

We consider the run-time behavior of a state-of-the-art tabu search algorithm for the JSP, which uses reintensification and path relinking in conjunction with a simple elite pool maintenance scheme. In the course of our analysis, we identify a number of algorithm "failure modes", i.e., behaviors exhibited by the algorithm that can prevent it from attaining even higher levels of performance. Where relevant, we further identify the underlying cause in terms of search space structure. Such analyses, although atypical of most metaheuristic research, serve two fundamental purposes. First, failure modes provide a target for the design of new metaheuristics, allowing research to move beyond the more typical unguided paradigm. Second, failure modes can potentially serve as the basis for a more detailed theory concerning the behavior of metaheuristics with long-term memory, the preliminaries of which are currently unavailable. Due to the compressed nature of the presentation, we focus on illustrative, concrete examples of each of the failure modes we discuss, and limit our presentation to three specific failure modes. A more rigorous, statistical documentation of the observed phenomena are provided in an expanded sequel, where additional failure modes are also discussed.

*Sandia is a multipurpose laboratory operated by Sandia Corporation, a Lockheed-Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.

2 Tabu Search and the Job-Shop Scheduling Problem

We consider the *NP*-hard job-shop scheduling problem (JSP) with the makespan-minimization objective [1]. Although a large number of algorithmic paradigms (including nearly all metaheuristics) have been applied to the JSP, tabu search algorithms have dominated all others in terms of both performance and solution quality since the late 1980s [4]. The current state-of-the-art tabu search algorithm for the JSP, denoted *i*-TSAB, was introduced by Nowicki and Smutnicki in 2001 [7]. Based on their earlier, highly successful TSAB tabu search algorithm [6], *i*-TSAB employs a small pool of elite solutions from which both reintensification and path relinking are periodically applied. Recently, we performed a rigorous analysis of *i*-TSAB, with the goal of identifying those specific components that enable it to achieve state-of-the-art performance levels [9]. As a result, we were able to develop a simplified version of *i*-TSAB, which we denote *i*-STS, that is competitive with *i*-TSAB but is more amenable to analysis.

Search in *i*-STS is based on a small pool of elite solutions P; in our experiments, $|P| = 8$. Mirroring *i*-TSAB, *i*-STS proceeds in two phases: the *initiation* phase and the *proper work* phase. In the initiation phase, the elite pool P is populated by performing moderate-length runs of a core tabu search from randomly and independently generated local optima; the best solution found during the course of each run is inserted into P. In the proper work phase, reintensification and path relinking procedures are applied to the solutions in P, with respective probabilities of p_i and p_d. Under reintensification, a short run of tabu search is performed from a random element $p \in P$ of the elite pool. Upon termination, if the makespan of the best solution s^* found during search is lower than that of p, s^* replaces p in P. Under diversification, path relinking is performed between two randomly selected solutions $p, q \in P$ to yield a solution s that is roughly equi-distant from both p and q. A short run of tabu search is performed from s, yielding a best-during-run solution s^*. If the makespan of s^* is lower than that of p (i.e., a randomly selected parent), then s^* replaces p in P. The process continues for a user-specified number of total iterations of the core tabu search procedure. Full details of the *i*-STS algorithm, including performance results relative to *i*-TSAB on Taillard's benchmark problems are provided in [9].

In the remaining sections, we make extensive use of the *distance* between two solutions s_1 and s_2, which we take as the well-known disjunctive graph distance [5]. Although the full details are beyond the scope of the current presentation, we simply note that the disjunctive graph distance provides a lower bound on the number of moves required under widely used local search move operators to transform s_1 into s_2, i.e., the measure is directly related to the behavior of *i*-STS. Much of the analysis is focused specifically on a set of extremely difficult 20-job, 20-machine benchmark instances introduced by Taillard [8], which are denoted `ta21` through `ta30`. All of these instances are "open", in that the optimal makespans are unknown. However, through extensive experimentation, we have obtained large sets (of at least 10 elements apiece) containing the best-known solutions to these instances; we observe that the such solutions are not unique. We use these solution sets to perform *a posteriori* analyses of *i*-STS runs that failed to locate a best-known solution, with the goal of understanding why the higher performance level was not achieved. Finally, we also use a number of smaller 10-job, 10-machine benchmark instances in situations where the analysis necessitates knowledge of the optimal makespan.

3 Why Reintensification Can Fail

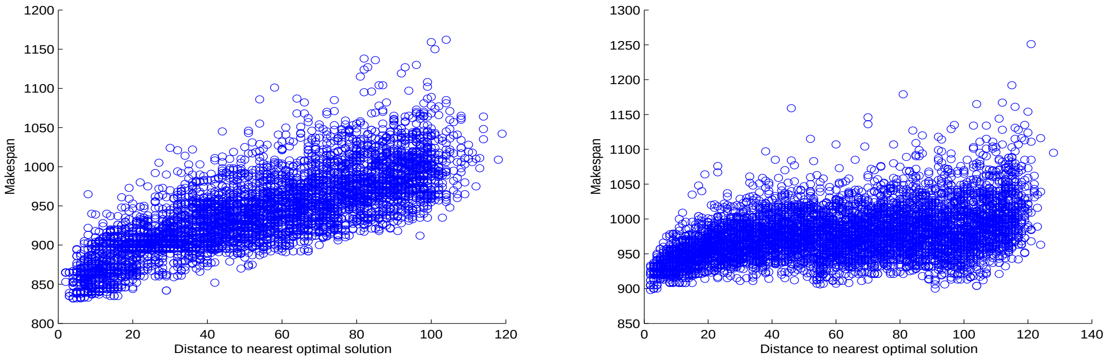

Figure 1: Examples of 10-job, 10-machine JSP instances that exhibit high (left figure) and low (right figure) fitness-distance correlation

It is generally accepted that a careful balance between intensification and diversification is required to achieve high-performance strategies based on pool maintenance schemes [3]. However, we have a poor understanding of the circumstances under which reintensification can fail, especially in terms of the relationship between search space structure and algorithm run-time dynamics. Several researchers have argued that fitness-distance correlation (i.e., good solutions are clustered in restricted regions of the search space) is relatively high in the JSP [7]. In the left-hand side of Figure 1, we show a scatter-plot of the distance from a solution to the nearest optimal solution versus solution makespan, for one such 10-job, 10-machine instance. We have previously demonstrated that tabu search in the JSP is actually biased *away* from optimal solutions [10]. In the context of the instance shown in the left-hand side of Figure 1, this implies that search is empirically concentrated in the region of solutions that are roughly distance 40 to 80 away from the nearest optimal solution. Thus, if tabu search is initiated from solutions that are proximate to optimal solutions, the probability of locating an optimal solution quickly drops as the number of iterations increases. In this context, periodic reintensification around high-quality solutions is an effective strategy. High-quality solutions are clustered and are consistently near even higher-quality solutions. Consequently, reintensification is likely to both (1) avoid extensive search in the region of average-quality solutions distance 40-80 from the nearest optimal solution and (2) locate higher-quality solutions with relatively high probability.

Next, consider the impact of reintensification when fitness-distance correlation is low, e.g., for the instance shown in the right side of Figure 1. Here, we observe two clusters of near-optimal solutions in the regions 55-65 and 85-100 distance away from the nearest optimal solution. Further, because tabu search concentrates search effort in the region distance 40-80 away from the nearest optimal solution, the near-optimal solutions at distances 55-65 are more likely to be encountered than those near-optimal solutions at distance 1-10. Thus, under a pool maintenance scheme coupled strictly with a reintensification mechanism, (1) elite pool elements are likely to be very far from optimal solutions and (2) intensification - because of the limited duration - is unlikely to ever reach an optimal solution. In the sequel to this abstract, we describe in complete detail a series of experiments in which we conclusively demonstrate a strong correlation between instance fitness-distance correlation and the efficacy of reintensification: reintensification only fails when there exist clusters of high-quality solutions that are very distant from the set of optimal solutions. Finally, in contrast to [7], we observe that roughly half of Taillard's 20-job, 20-machine instances exhibit very poor fitness-distance correlation, e.g., similar to that shown in the right side of Figure 1.

4 When Reintensification Becomes Diversification

Reintensification in i-STS simply selects an elite solution $p \in P$ at random, and applies a core tabu search procedure for a fixed number of iterations; additional details include randomization of the initial moves to force search along different trajectories. For 20-job, 20-machine JSP instances, i-STS and i-TSAB both allocate a minimum of 7,000 iterations to each reintensification attempt. The implicit assumption in this allocation is that search is likely to remain nearby the initial solution $p \in P$, i.e., that intensification is actually being performed. Recall from Section 3 that tabu search in the JSP is actually biased away from optimal solutions. An analogous observation holds for arbitrary solutions, i.e., the search space is structured in such a manner that tabu search is likely to quickly move away from any given fixed solution. This observation leads us to hypothesize that much of what is labeled as "intensification" may in fact be acting as a diversification mechanism.

Consider Taillard's 20-job, 20-machine benchmark instances, in which the maximal possible disjunctive graph distance between any two solutions is 3800. The mean distance between random local optima is roughly 500; the latter can be viewed as a measure of the "radius" of the search space. The large difference relative to the theoretical maximum is due to the large number of infeasible solutions in the search space. Over 10 trials of i-STS (duration 50 million iterations of tabu search apiece) on each of these problem instances, reintensification is on average at least distance 100 from the initial solution p after only 500 iterations. After 2,500 iterations, reintensification is on average at least distance 250 from the initial solution p. Further, the distance is proportional to the number of iterations expended. Thus, after roughly 1/3 of the total allocation of iterations, search has traversed at least half of the distance between a typical pair of random optima. The implication of this observation is obvious: a very significant proportion of the search effort that is labeled as "intensification" is actually a strong form of diversification. We make no claim that the mechanism is ineffective. Rather, we simply observe that the mechanism exhibits behavior well beyond the original intent of its design. By not validating hypotheses underlying mechanism design, we run the real risk of propagating incorrect beliefs regarding the behavior of metaheuristics, which ultimately hinders scientific progress in the field.

5 Are Metaheuristics Effective Intensifiers?

The central role of reintensification in many metaheuristics, including i-STS, suggests that the effectiveness of metaheuristics as intensification mechanisms is well established. However, we have developed concrete experimental evidence that is contrary to this point of view, at least in the JSP. For each of Taillard's 20-job, 20-machine instances, we execute 10 independent trials of i-STS for a total of 50 million iterations of the core tabu search procedure. At the end of each trial, we record the contents of the elite pool P. Using the set B of best-known solutions for each instance (described in Section 2), we then compute the distance between each $p \in P$ and the closest (in terms of disjunctive graph distance) solution $b \in B$. Although descriptive in nature, the intent behind this analysis is to test the following hypothesis: when i-STS fails, it is because the elite solutions in P are very distant from the best-known or "target" solutions. Initially, we suspected that search in i-STS was becoming trapped in high-quality but sub-optimal regions of the search space, and diversification in the form of path relinking failed to allow search to progress from solutions in P to those far away in B. While we did observe this condition, it was not dominant in those trials that failed to find a best-known solution. Rather, at least one of the elite solutions in P were often surprisingly close to solutions in B – but the reintensification mechanism was failing to locate such solutions.

For an illustrative example, consider the `ta23` benchmark instance, with a best-known makespan equal to 1557. For this and other 20-job, 20-machine benchmark instances, the mean disjunctive graph distance between random local optima is roughly 500; the maximum possible distance is 3800. Only two of the ten i-STS trials located a solution with a makespan equal to the best known. For the remaining trials, the worst solution obtained yielded a makespan equal to 1561. The question is then: Why is i-STS failing to locate solutions with 1557 makespan? Unexpectedly, our analysis indicates that in these "failed" trials, at least one of the solutions in the final elite pool P is within at *most* distance 50 of a solution with 1557 makespan. In other words, i-STS has located the salient region of the search space, but reintensification via the core tabu search procedure is failing to locate the best solution within this sub-space. This phenomena is not due to the deterministic nature of tabu search (we randomize both the initial moves and employ random tie-breaking in i-STS), and we have observed identical behaviors using more stochastic metaheuristics, including iterated local search. Further, the region is typically located very early in execution, implying the phenomenon is not due to the lack of intensification attempts. Qualitatively identical behavior is observed on many of Taillard's 20-job, 20-machine instances, most prominently on `ta23`, `ta24`, `ta27`, `ta29`, and `ta30`. Finally, although the differences in performance cited in the above example may appear minimal, such magnitudes are sufficient to distinguish state-of-the-art performance from second-tier competition.

This finding suggests that metaheuristics are *not* necessarily effective intensification mechanisms. Given straightforward numerical calculations, this is not surprising *a posteriori*; even restricting search to solutions distance 50 away from a reference point yields a combinatorially vast sub-space. Clearly, more research is needed into the development of intensification mechanisms for metaheuristics. In particular, more exhaustive or partially enumerative mechanisms seem like a promising alternative, for example.

6 Conclusions

Via detailed, *a posteriori* analyses of algorithmic runs, we have identified a number of failure modes that often prevent i-STS, a state-of-the-art tabu search algorithm for the JSP, from consistently locating solutions with makespans equivalent to the best known. These failure modes (1) expose a serious design flaw in i-STS, (2) invalidate a well-known hypothesis regarding the effectiveness of metaheuristics as intensifiers, and (3) provides new insight into the relationship between pool maintenance, long-term memory, and search space structure. One obvious and immediate use of these results is to develop improved variants of i-STS. We have already initiated this research, and preliminary results are extremely promising, the presentation of which is omitted here due to our focus on algorithm analysis. Longer-term, these and other failure modes will ideally serve to develop a concrete theory of pool-based metaheuristics; the preliminary insights we have gained provide a first step in this direction. Finally, it will be illustrative to contrast failure modes observed for metaheuristics for the JSP with those in other problems, to gain insight into the potential universality of advanced metaheuristic design concepts.

References

[1] Błażewicz, J., and Domschke, W., and Pesch, E (1996): “The Job Shop Scheduling Problem: Conventional and New Solution Techniques”. In: *European Journal of Operational Research* **93** (1), 1–33.

[2] Glover, F., and Laguna, M. (1997): *Tabu Search*. Kluwer Academic Publishers, Boston, USA.

[3] Greistorfer, P (2005): “Controlled Pool Maintenance in Combinatorial Optimization”. In: Rego, C., and Alidaee, B. (eds.): *Metaheuristic Optimization via Memory and Evolution*, Kluwer Academic Publishers, Boston, USA.

[4] Jain, A.S., and Meeran, S (1999): “Deterministic job-shop scheduling: Past, present, and future”. In: *European Journal of Operational Research* **113** (2), 390–434.

[5] Mattfeld, D.C., and Bierwirth, C., and Kopfer, H. (1999): “A Search Space Analysis of the Job Shop Scheduling Problem”. In: *Annals of Operations Research* **86**, 441-453.

[6] Nowicki, E., and Smutnicki, C. (1996): “A Fast Taboo Search Algorithm for the Job Shop Problem”. In: *Management Science* **42** (6), 797–813.

[7] Nowicki, E., and Smutnicki, C. (2005): “Some New Ideas in TS for Job Shop Scheduling”. In: Rego, C., and Alidaee, B. (eds.): *Metaheuristic Optimization Via Memory and Evolution*. Kluwer Academic Publishers, Boston, USA, 165–190.

[8] Taillard, É.D. (1993): “Benchmarks for Basic Scheduling Problems”. In: *European Journal of Operational Research* **64** (2), 278–285.

[9] Watson, J.P., and Howe, A.E., and Whitley, L.D. (Submitted): “Deconstructing Nowicki and Smutnicki's i-TSAB Tabu Search Algorithm for the Job-Shop Scheduling Problem”.

[10] Watson, J.P., and Whitley, L.D., and Howe, A.E. (2003): “A Dynamic Model of Tabu Search for the Job-Shop Scheduling Problem”. In: *Proceedings of the First Multidisciplinary International Conference on Scheduling: Theory and Applications.*